

What is claimed is:

1. A method of lexical analysis, comprising:
receiving an input stream of characters;
5 detecting a delimiter in the input stream, the delimiter being
selected from the group consisting of a single character delimiter and a
multi-character delimiter; and
returning a token upon detecting the delimiter.
- 10 2. The method of claim 1, further comprising:
reading the input stream one character at a time.
3. The method of claim 1, further comprising:
forming the token by appending to a string at least one of the input
15 stream characters preceding the delimiter.
4. The method of claim 1, further comprising:
detecting a delimiter-token;
returning the token upon detecting the delimiter token.
- 20 5. The method of claim 4, further comprising:
returning the delimiter-token.
6. The method of claim 5, wherein the delimiter-token is returned on a
25 subsequent call to a lexical analyzer.

7. The method of claim 1, wherein the step of detecting includes:
comparing at least one of the input stream characters to a single
character delimiter table and a multiple character delimiter table.

5

8. The method of claim 1, for use in migrating pre-existing software
code from a first version to a second version of a predetermined language.

10

9. A lexical analyzer, comprising:
an input for receiving an input stream of characters;
a detector for detecting a delimiter in the input stream, the delimiter
being selected from the group consisting of a single character delimiter and a
multi-character delimiter; and
an output for returning a token upon detecting the delimiter.

15

10. The lexical analyzer of claim 9, further comprising:
means for forming the token by appending to a string at least one
of the input stream characters preceding the delimiter.

20

11. The lexical analyzer of claim 9, further comprising:
means for detecting a delimiter-token;
means for returning the token upon detecting the delimiter-token.

25

12. The lexical analyzer of claim 11, wherein the delimiter-token is
returned on a subsequent call to the lexical analyzer.

T06230" 66402860

13. The lexical analyzer of claim 9, wherein the detector includes:
a comparator for comparing at least one of the input stream
characters to a single-character delimiter table and a multiple-character delimiter
5 table.

14. The lexical analyzer of claim 9, for use in migrating pre-existing
software code from a first version to a second version of a predetermined
language.
10

15. Computer program product in a computer-usable medium,
comprising:
means for receiving an input stream of characters;
means for detecting a delimiter in the input stream, the delimiter
15 being selected from the group consisting of a single character delimiter and a
multi-character delimiter; and
means for returning a token upon detecting the delimiter.

16. The computer program product of claim 9, further comprising:
20 means for forming the token by appending to a string at least one
of the input stream characters preceding the delimiter.

17. The computer program product of claim 9, further comprising:
means for detecting a delimiter-token; and
25 means for returning the token upon detecting the delimiter-token.

1098049-03901
"03901" 03901

18. The computer program product of claim 11, wherein the delimiter-token is returned on a subsequent call to the computer program product.

5

19. The computer program product of claim 9, wherein the detector includes:

means for comparing at least one of the input stream characters to a single-character delimiter table and a multiple-character delimiter table.

10

20. The computer program product of claim 9, for use in migrating pre-existing software code from a first version to a second version of a predetermined language.

T063E0" 66402860